

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-324000

(43)Date of publication of application : 08.11.2002

(51)Int.Cl. G06F 12/00
G06F 3/06

(21)Application number : 2002-003193

(71)Applicant : HITACHI LTD

(22)Date of filing : 10.01.2002

(72)Inventor : WATANABE CHIYOKUKI
YAMAMOTO AKIRA

(30)Priority

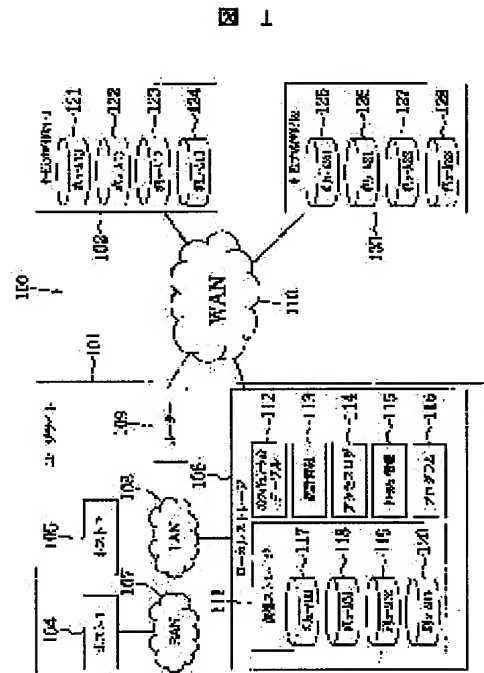
Priority number : 2001 760344 Priority date : 12.01.2001 Priority country : US

(54) VIRTUAL VOLUME STORAGE

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a system for providing a data storage service.

SOLUTION: This system comprises service provider sites 102 and 103 configured to provide the data storage service, and a user site 101 coupled by a WAN(wide area network) 110 to the service provider site. The user site is provided with a local storage 106 having a virtual storage 111, the virtual storage is provided with a synchronous volume and an asynchronous volume, and the local storage is configured to immediately transmit to the data service provider data that is written in the synchronous volume, to transmit at a predetermined schedule to the service provider site data that is written in the asynchronous volume, and to read data from the service provider site if the data is not stored in the local storage.



特選2002-324000

(P2002-324000A)

(43)公開日 平成14年11月8日(2002.11.8)

(51)Int.Cl. ⁷	識別記号	F I	予選ノット(参加)
G 0 6 F 12/00	5 1 4	G 0 6 F 12/00	5 1 4 E 5 B 0 6 5
	5 3 1		5 3 1 D 5 B 0 8 2
	5 4 5		5 4 5 A
3/06	3 0 4	3/06	3 0 4 F

審査請求 未請求 請求項の数17 O L 外国語出題 (全54頁)

(21) 出版番号	特観2002-3193(F2002-3193)	(71) 出版人	000005108 株式会社日立製作所
(22) 出版日	平成14年1月10日(2002.1.10)	(72) 発明者	渡辺 直金 アメリカ合衆国 カリフォルニア州 プリ アメルン シェラポインパークウェイ 2000 日立アメリカ リサーチアンドディ ベロップメント部内
(31) 優先権主張番号	09/760344	(73) 発明者	山本 彰 神奈川県川崎市麻生区王禅寺1093番地 株 式会社日立製作所システム研究研究所内
(32) 優先日	平成13年1月12日(2001.1.12)	(74) 代理人	100091086 弁護士 平木 祐輔
(33) 優先権主張国	米国 (US)		

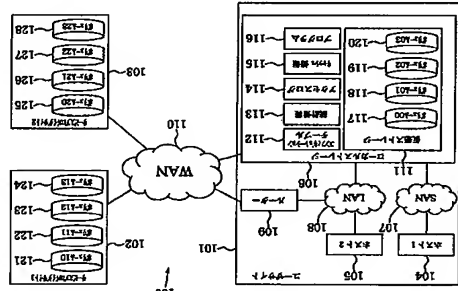
最終頁に続く

(54)【発明の名称】 仮想ボリューム・ストレージ

(57)【要約】

【課題】 データ・ストレージ・サービスを提供するシステムを提供する。

【解説】 データ・ストレージ・サービスを提供するために構成されているサービス・プロバイダ・サイトは、IN2、103及びサービス・プロバイダ・サイト1W、AN11.10により統合されたユニザ・サイト101を備える。ユニザ・サイトは仮想ストレージ111を支持する。データ・ストレージ108を備え、仮想ストレージは同期ポリユーニザ及び非同期ポリユーニザを備え、ローカル・ストレージ109を備え、仮想ストレージは同期ポリユーニザ及び非同期ポリユーニザに書き込まれたデータを、非同期ポリユーニザに書き込まれたデータを所定のスケジューリングでサービス・プロバイダ・サイトに送信し、データがローカル・ストレージに格納されていない場合はサービス・プロバイダ・サイトからデータを読み取るように構成されている。



8
・プロバイダはユーザ・アクセス活動のパターン分析で
きる。この分析に基づき、サービス・プロバイダはユー
ザ・サイトの仮想ポリューム・ストレージをチェーニ
ングできる。

【0016】

【発明の実施形態】以下では、当業者が発明を実施
し使用できるように説明し、また特定のアプリケーション
とその要件の文脈において説明を行っている。実施形
態のさまざまな修正は、当業者であればやすく理解で
きるであろうし、またここで定義している一般的な原理
は本発明の精神と範囲を逸脱することなく他の実施形態
で示されている実施形態に限定されるのではなく、ここ
で開示されている原理、特徴および教示に一致する最も
広い範囲を許容するものとする。

【0017】本発明による一実施形態では、システム1
00はユーザ・サイト101、さらに第1のサービス・プロ
バイダ・サイト102及び第2のサービス・プロバイ
ダ・サイト103により構成される。ユーザ・サイト
101の詳細を以下で説明する。システム100内のサ
ービス・プロバイダ・サイトの数は異なっているも
よ。第1のサービス・プロバイダ・サイト102は、ス
トレージ・ポリューム121、122、123、及び1
24を備え、第2のサービス・プロバイダ・サイト10
3はストレージ・ポリューム125、126、127、
及び128を備える。サービス・プロバイダ・サイト内
のストレージ・ポリュームの数は異なっているもよ。
サービス・プロバイダ・サイト102及び103は、た
えば、2つの異なる安全な建物内に配置できる。ユー
ザ・サイト101、第1のサービス・プロバイダ・サイ
ト102、及び第2のサービス・プロバイダ・サイト1
03はそれぞれ、ワイド・エリア・ネットワーク (WA
N) 110に接続されている。サービス・プロバイダ1
02及び103は、WAN 110経由でデータ・スト
レージ・サービスをユーザ・サイト101のユーザに提
供する。

【0018】当業者には周知のとおり、WANとは、通
常比較的広い範囲にわたる地理的領域にまたがるコンピ
ュータ・ネットワークのことである。WANはまた、ロー
ーカル・エリア・ネットワーク (LAN) も含む。WAN
Nに接続されているコンピュータは、多くの場合、電話
システムなどの公衆網を通じて接続される。さらに、専
用回線や衛星を通じて接続することもできる。現存最大
のWANはインターネットである。

【0019】WAN 110では、各サイト (ユーザ・
サイト101とサービス・プロバイダ・サイト102及
び103) の間で通信ができる。各サイト101、10
2、及び103は、長い距離で互いに隔てられていてよ
い。WAN 110は、通常、非同期転送モード (AT
M)、同期光ネットワーク (SONET)、高密度波長

9
分割多重 (DWDM)、又はインターネット・プロトコ
ル (IP) ネットワークにより構成される。

【0020】ユーザ・サイト101では、第1のホスト
104はストレージ・エリア・ネットワーク (SAN)
107を介してローカル・ストレージ106に接続さ
れ、第2のホスト105はローカル・エリア・ネットワ
ーク (LAN) 108を介してローカル・ストレージ1
06に接続されている。ホスト104及び105はロー
カル・ストレージ106にアクセスする。ホスト104
及び105は、たとえば、サーバである。ユーザ・サイ
ト101内のホストの数は異なっているもよ。LAN
108は、たとえば、ルータ109を介してWAN
110に接続されている。本発明の教示から、SAN
107又はLAN 108のいずれか1つ、又はSAN
107及びLAN 108の両方を含むようにユーザ
・サイト101を実装することができ、これを当業者は
理解するであろう。

【0021】当業者には周知のように、SANは共有ス
トレージ・デバイスの高速サブネットワークであり、S
ANによりLAN又はWAN内のすべてのサーバからす
べてのストレージ・デバイスが利用可能である。SAN
にさらにストレージ・デバイスを追加すると、その追加
されたストレージ・デバイスも、さらに大きなネットワ
ーク内の任意のサーバからアクセスできるようになる。

【0022】SAN 107は、たとえば、ファイバ・
チャネル又は小型コンピュータ周辺機器インタフェー
ス (SCSI) で構成できる。当業者には周知のよう
に、ファイバ・チャネルはコンピュータ及び大容量記憶
デバイス製造業者のコンソシアムが開発したシリアル
・データ転送アーキテクチャであり、現在は、米国規格
協会 (ANSI) によって標準化されている。最も有名
なファイバ・チャネル標準は、Fibre Channel Arbitrat
ed Loop (FC-A-L) で、新しい大容量記憶デバイス
や非常に高い帯域幅を必要とするその他の周辺機器用に
設計されている。光ファイバでデバイスを接続するFC
-A-Lは、全二重データ転送を行い、転送速度は毎秒約
100メガバイト (MBps) である。

【0023】また当業者には周知のように、SCSIは
種々のコンピュータシステムで周辺装置をコンピュータ
に接続するために使用されているパラレル・インタフェ
ース標準である。SCSIインタフェースは、標準装備
のシリアル・ポート及びパラレル・ポートに比べて高速
なデータ転送速度 (最大約80MBps) を利用でき
る。

【0024】また当業者には周知のように、LANと
は、通常比較的狭い地域に置かれるコンピュータ・ネッ
トワークのことである。ほとんどのLANは、単一の建
物又は建物群に制限されている。ほとんどのLANは、バ
ックステージーション及びパベソナル・コンピュータを接
続する。LAN内の各ノード (個々のコンピュータ)

10
は、個々に中央処理装置 (CPU) を備え、プログラム
を実行するが、LAN上の任意の場所にあるデータ及び
デバイスにアクセスすることも可能である。したがっ
て、多くのユーザが、レーザ・プリンタなどの周辺デ
バイスをデータとともに共有できる。ユーザはさらに、
LANを使用し、たとえば電子メールを送信したりチ
ャット・セッションに参加したりして互いに通信するこ
ともできる。LANにはいろいろな種類のものがある
が、PCにはイーサネット (登録商標) が最も一般的で
ある。LANは非常に高速にデータを送信することがで
き、電話回線でデータを送信するよりもかなり高速であ
る。しかし、LANの配線距離は制限されており、また
単一のLANに接続できるコンピュータの台数にも制限
がある。

【0025】図1では、LAN 108をたとえばイー
サネット (登録商標) として構成している。第1のホス
ト104は、たとえば、ブロック (SCSI) インタフ
ース (I/F) を使用してローカル・ストレージ10
6にアクセスする。第2のホスト105は、たとえば、
ファイル (NFS又はCIFS) I/Fを使用してロー
カル・ストレージ106にアクセスする。第2のホスト
105は、たとえば、ブロック (iSCSI) I/Fを
使用してローカル・ストレージ106にアクセスするこ
ともできる。

【0026】ローカル・ストレージ106及びサービス
・プロバイダ・サイト102及び103では、ATML
でiSCSI又はファイバ・チャネル、SONET上で
ファイバ・チャネル、又は独自ベンダ・プロトコルを使
用できる。ローカル・ストレージ106は、仮想スト
レージ111、コンプライゼーション・テーブル112
(図4も参照)、統計情報113 (図5も参照)、アク
セス・ログ114 (図6も参照)、キャッシュ情報11
5、及びプログラム116を含む。コンポーネント11
2、113、114、115、及び116ではローカル
・ストレージ106をキャッシュとして動作させること
ができる。ホスト104及び105は仮想ストレージ1
11にアクセスできる。仮想ストレージ111は、いく
つかのポリューム117、118、119、及び120
を含む。仮想ストレージ111内のポリュームの数は異
なっているもよ。これらのポリューム117-120
の管理は、ローカル・ストレージ106が行う。ローカ
ル・ストレージ106及びサービス・プロバイダ・サイ
ト102及び103は、仮想ポリューム・ストレージ1
11の作成のための共同作業を行う。

【0027】ユーザは、ユーザ・サイト101に仮想ポ
リューム・ストレージ111 (ローカル・ストレージ1
06内) を配属することができる。仮想ポリューム・
ストレージ111がユーザ・サイト101に配属される
と、ホスト104及び/又はホスト105のユーザは仮
想ポリューム・ストレージ111を使用することによ

り、巨大なサービス・プロバイダ・ストレージ (ボリューム121~124及び/又はボリューム125~128) にアクセスできる。仮想ボリューム125~128は通常、サービス・プロバイダ・ストレージ・ボリューム (ボリューム121~124及び/又はボリューム125~128) に比べてサイズが小さい。仮想ボリューム11は、たとえば以下で説明する。[0031] 保留データ・リスト211は、ローカル・ストレージ106に保存されている保留データのリストである。保留データ・リスト211は、保留データへのポインタ、サービス・プロバイダ・サイトへのポインタ、及び同期期間などの各保留データの情報を含む。同期期間は、データの特性により定義される。データが重要でない場合、このデータは所定の期間、ローカル・ストレージ106内に格納することができる。この期間は、たとえば、1分、1時間、1日、1週間、又は1カ月とすることができる。データが重要な場合、データが仮想ストレージ111に格納された後、コンフィギュレーション・テーブル112 (図4) を使用してデータを (同期して) サービス・プロバイダ・サイト (たとえば、サイト102) に格納すべきである。たとえば、同期データはボリューム00 117に格納される。図4で、ボリューム00 117は識別番号ID 00で識別される。ボリューム00 117に格納されているデータは同期データなので、データはサービス・プロバイダ・サイト102 (リモート・サイト1) のボリューム10 121 (リモートID 10) に即座に格納される。

[0032] この同期アクセス機能を使用すると、信頼性の高いデータ・ストレージを構築できる。ただし、ユーザー・サイト101からサービス・プロバイダ・サイト102及び/又は103にアクセスするのに時間がかかると、同期アクセスにはかなりの時間が必要である。必要なアクセス時間は、システム100で実行される特定のオペレーションによって異なる。

[0033] ホスト104及び/又は105からアクセスされたデータはメモリ207に格納される。このデータは、キャッシュ・データ212と呼ばれる。キャッシュ・データ212は、ホスト104及び/又はホスト105のユーザーによって構築にアクセスされる一時的データである。

[0034] 一実施形態では、プログラム116は、サービス・プログラム213、シンクロナイザ217、キャッシュ制御218、スケジューラ220、及びデバイス・ドライバ219を含む。サービス・プログラム213では、ローカル・ストレージ106及びホスト104 (及び/又はホスト105) が互いに通信できる。サービス・プログラム213は、NFSサーバ214、CIFSサーバ215、及びSCSIサーバ216で構成される。NFSサーバ214は、従来のNFSサーバとして動作する。CIFSサーバ215、従来のCIFSサーバとして動作する。SCSIサーバ216は、SCSIデータ

ゲット・デバイスとして動作する。シンクロナイザ217は、保留データ・リスト211を使用して、ローカル・ストレージ106とサービス・プロバイダ・サイト102及び/又は103の同期をとる。キャッシュ制御218は、キャッシュ・ディレクトリ210を使用して、メモリ207とローカル・ディスク209内のキャッシュ・データ212を制御する。スケジューラ220は、キャッシュ制御218、及びデバイス・ドライバ219などのすべてのプロセス・ジョブをスケジューリングする。デバイス・ドライバ219は、SAN I/F 203、LAN I/F 204、WAN I/F 205、メモリ1/F 206、及びディスク・コントローラ208などのローカル・ストレージ106内のすべてのデバイスを制御する。

[0035] 図3は、本発明の実施形態によるデータ・レイアウトの一例を示すブロック図である。ローカル・ストレージ106の管理は、以下でさらに詳しく説明しているように、データの特徴を使用することに基づいている。仮想ボリューム・ストレージ111では、データが静的にキャッシュされているか、同期が非同期かに応じてデータの特徴を取り扱う。以下でさらに詳しく説明しているように、仮想ボリューム・ストレージ111は、ストレージ・ボリュームだけでなく、ディレクトリ、ファイル、シリンダ、及び/又はブロック・アドレスでもデータの特徴を取り扱う。ユーザー・サイト101では、仮想ストレージ111に4種類の仮想ボリューム (ボリューム00 117、ボリューム01 118、ボリューム02 119、及びボリューム03 120) がある。これらの仮想ボリュームの特徴である、(1) 静的同期ボリューム、(2) 静的非同期ボリューム、(3) キャッシュされた同期ボリューム、及び(4) キャッシュされた非同期ボリュームについて次に説明する。

[0036] (1) 静的同期ボリューム (ボリューム00 117): ボリューム00 117は静的同期ボリュームである。したがって、ボリューム00 117は、第1のサービス・プロバイダ・サイト102において、実際のストレージ・ボリューム106内に入った後、ローカル・ストレージ106はメモリ207を介してローカル・ディスク209からデータをホスト104及び/又は105に送る。アクセスされたデータがキャッシュ・データ212内にある場合、ローカル・ストレージ106はローカル・ディスク209にアクセスせず、メモリ207にアクセスするだけである。ホスト104及び/又は105は、ボリューム02 119とボリューム12123にデータを書き込む。このキャッシュされた同期ボリューム、ボリューム02 119は都合よく、ローカル・ストレージ106のサイズを減らすことができる。この仮想ボリューム、ボリューム02 119ではさらに、信頼性の高いストレージ・システムを構築できる。

スと信頼性の高いストレージ・システムを実現する。[0037] (2) 静的非同期ボリューム (ボリューム01 118): ボリューム01 118は静的非同期ボリュームである。したがって、ボリューム01 118は、第1のサービス・プロバイダ・サイト102において実際のストレージ・ボリュームであるホスト104及び/又は105は、ボリューム01 118からデータを読み込み、ボリューム11 122からデータを読み込まない。アクセスされたデータがキャッシュ・データ212内にある場合、ローカル・ストレージ106はローカル・ディスク209にアクセスせず、メモリ207にアクセスするだけである。ホスト104及び/又は105は、ボリューム01 118にデータを書き込み、保留データ・リスト211 (図2) に登録する。この保留データは、以下で説明するように、所定のスケジュールで同期処理される。

[0038] この保留データは、バックグラウンド・ジョブでローカル・ストレージ106 (図2) 内に格納し、WAN 110のトラフィックを低減することができる。この静的非同期ボリューム、ボリューム01 118は、ボリュームの高速な読み書きを行うが、信頼度は同期ボリュームの場合と同じではないことがある。

[0039] (3) キャッシュされた同期ボリューム (ボリューム02 119): ボリューム02 119はキャッシュされた同期ボリュームである。したがって、ボリューム02 119は、第1のサービス・プロバイダ・サイト102において実際のストレージ・ボリュームであるボリューム12 123と同じサイズを占める。ボリューム02 119はキャッシュとして動作する。したがって、ホスト104及び/又は105によってアクセスされるデータがボリューム02 119内にある、アクセスされたデータがローカル・ディスク209 (図2) 内にある場合、ローカル・ストレージ106 (図2) は第1のサービス・プロバイダ・サイト102からデータを読み込んで、読み込んだデータをローカル・ディスク209に書き込む。データがローカル・ストレージ106内に入った後、ローカル・ストレージ106はメモリ207を介してローカル・ディスク209からデータをホスト104及び/又は105に送る。アクセスされたデータがキャッシュ・データ212内にある場合、ローカル・ストレージ106はローカル・ディスク209にアクセスせず、メモリ207にアクセスするだけである。ホスト104及び/又は105は、ボリューム02 119とボリューム12123にデータを書き込む。このキャッシュされた同期ボリューム、ボリューム02 119は都合よく、ローカル・ストレージ106のサイズを減らすことができる。この仮想ボリューム、ボリューム02 119ではさらに、信頼性の高いストレージ・システムを構築できる。

り、巨大なサービス・プロバイダ・ストレージ (ボリューム121~124及び/又はボリューム125~128) にアクセスできる。仮想ボリューム125~128は通常、サービス・プロバイダ・ストレージ・ボリューム (ボリューム121~124及び/又はボリューム125~128) に比べてサイズが小さい。仮想ボリューム11は、たとえば以下で説明する。[0031] 保留データ・リスト211は、ローカル・ストレージ106に保存されている保留データのリストである。保留データ・リスト211は、保留データへのポインタ、サービス・プロバイダ・サイトへのポインタ、及び同期期間などの各保留データの情報を含む。同期期間は、データの特性により定義される。データが重要でない場合、このデータは所定の期間、ローカル・ストレージ106内に格納することができる。この期間は、たとえば、1分、1時間、1日、1週間、又は1カ月とすることができる。データが重要な場合、データが仮想ストレージ111に格納された後、コンフィギュレーション・テーブル112 (図4) を使用してデータを (同期して) サービス・プロバイダ・サイト (たとえば、サイト102) に格納すべきである。たとえば、同期データはボリューム00 117に格納される。図4で、ボリューム00 117は識別番号ID 00で識別される。ボリューム00 117に格納されているデータは同期データなので、データはサービス・プロバイダ・サイト102 (リモート・サイト1) のボリューム10 121 (リモートID 10) に即座に格納される。

[0032] この同期アクセス機能を使用すると、信頼性の高いデータ・ストレージを構築できる。ただし、ユーザー・サイト101からサービス・プロバイダ・サイト102及び/又は103にアクセスするのに時間がかかると、同期アクセスにはかなりの時間が必要である。必要なアクセス時間は、システム100で実行される特定のオペレーションによって異なる。

[0033] ホスト104及び/又は105からアクセスされたデータはメモリ207に格納される。このデータは、キャッシュ・データ212と呼ばれる。キャッシュ・データ212は、ホスト104及び/又はホスト105のユーザーによって構築にアクセスされる一時的データである。

[0034] 一実施形態では、プログラム116は、サービス・プログラム213、シンクロナイザ217、キャッシュ制御218、スケジューラ220、及びデバイス・ドライバ219を含む。サービス・プログラム213では、ローカル・ストレージ106及びホスト104 (及び/又はホスト105) が互いに通信できる。サービス・プログラム213は、NFSサーバ214、CIFSサーバ215、及びSCSIサーバ216で構成される。NFSサーバ214は、従来のNFSサーバとして動作する。CIFSサーバ215、従来のCIFSサーバとして動作する。SCSIサーバ216は、SCSIデータ

[0030] キャッシュ情報115は、キャッシュ・ディレクトリ210、保留データ・リスト211、及びキャッシュ・データ212で構成される。キャッシュ・ディレクトリ210は、メモリ207及びローカル・ディスク209内のキャッシュされたデータ212のディレクトリ (コンフィギュレーション情報) である。このキャッシュ・データ212は、LRU (Least Recently Used)

CSI用であれば、「ファイルID」はNullとすることもできる。「アドレス」及び「サイズ」は、アクセス・アドレス及びサイズを示す。

【0053】読み込みプロセス

読み込みプロシージャでは、キャッシュされたデバイス（ローカル・ストレージ106）はキャッシュとして動作する。ホスト104及び/又は105は、ローカル・ストレージ106にアクセスするが、これは、ローカル・ストレージ106の空き容量が大きいからである。また、

ず、ホスト（たとえば、ホスト104）又はホスト105は、LAN 108（図1）を介してNFS、CIFS、又はiSCSIプロトコルの読み込みコマンドを実行するか、又はSAN 107を介してSCSIプロトコルの読み込みコマンドを実行する。ローカル・ストレージ106は、LAN 1/F 204又はSAN 1/F 203（図2）を介して読み込みコマンドを受

からのこのコマンドを処理し、このコマンドをスケジューラ220に入れる。次にスケジューラ220は読み込

ログラム2113内において適切なサーバ(NFSサーバ2114、CIFサーバ2115、又はSCSIサーバ2116)に入れて読み込みコマンドを処理することができ、各サーバ・プログラム2113は、読み込みコマンド

2) 内にあるかどうかを調べる。すべてのデータ(読み込みコマンドで要求された)がキャッシュ・データ21

バス202とネットワーク・インタフェース (SAN I/F 203又はLAN I/F 204) を介して

す。他方、データ（読み込みコマンドで要求された）の全部又は一部がキャッシュ・データ212内にない場

ら、又はサービスピバイダ・サイト102又は103からキャッシュ・データ212に移動すべきである。

込みプロセスの流れ図である。ローカル・ストレージ106がホスト(たとえば、ホスト104又は105)から読み込まれる。

ージ106はデータのボリューム・タイプを調べる(ス
テップ701)。スケジューラ220は、ボリュームI

テーブル112内のボリューム・タイプ(たとえば、静的タイプ)を調べる。特に、スケジューラ220(図

ーム・タイプを判別し、また読み込みコマンドを処理する(取り扱う)サーベ・プログラム(サーバ214、215)を決定する。サーバ216は、サーバ214、215

は、コンフィギュレーション・テーブル112 (図4)

した)はキャッシュ・データ212から、読み込みコマンドを送ったホスト(たとえばホスト104又は105)に移動される。特に、キャッシュ制御218はキャッシュ・データ212からデータを適切なサーバ・プログラム(214、215、又は216)に移動し、適切なサーバ・プログラム(214、214、又は216)がデータを、読み込みコマンドを送ったホストに返す。

【0059】書き込みプロセス

書き込みプロシージャでは、キャッシュされたデバイス（ローカル・ストレージ106）はキャッシュとして動作する。ホスト104及び/又は105は、ローカル・ストレージ106にアクセスするが、これは、ローカル・ストレージ106の空き容量が大きいからである。ま

ず、ホスト(たとえば、ホスト104又はホスト105)は、LAN 108を介してNFS、CIFS、又はiSCSIプロトコルの書き込みコマンドを発行する。

か、又はSAN 107を介してSCSIプロトコルの書き込みコマンドを発行する。ローカル・ストレージ106は、LAN 1/F 204又はSAN 1/F

・ドライバ219 (図2) は、ホストからのこの読み込みコマンドを処理し、このコマンドをスケジューラ222を介して書き込みコマンドを受信する。デバイス203

0)に入れる。次にスケジューラ220は書き込みコマンドを分析して、書き込みコマンドをサーバ・プログラム213内において適切なサーバ(NFSサーバ214、

CIFSサーバ215、又はSCSIサーバ216)に入れる。各サーバ・プログラム213は、書き込みコマンドのデータがキャッシュ・データ212(図2)内に

あるかどうかを調べる。すべてのデータ（書き込みコマンド）がキャッシュ・データ212内にある場合は、ローカル・ストレージ106は内部バス202とネット

ワーク・インタフェース (SAN I/F 203) は LAN I/F (204) を介してホストからデータを受信する。ローカル・ストレージ 106 は、ローカル・

ディスク209及びキャッシュ212内の割り当てられた空き領域にデータを上書きする。他方、(書き込みコマンドの)データの全部又は一部がキャッシュ・データマンドの)

212内にない場合、ローカル・ストレージ106は、キャッシュ・データ212内ないデータの残りについてある領域（ローカル・ディスク209及びキャッシュ

・データ212内の)を割り当てる。すべてのデータ領域の割り当てが済んだら、ローカル・ストレージ106はその割り当てられた領域内にデータを格納する。すべて

でのデータをキャッシュ・データ212内に格納する場合、ローカル・ストレージ106はローカル・ディスク209にデータを格納し、このデータをプロバイダ・サ

イト(たとえば、サービス・プロバイダ・サイト10
2)に送信する。

プロセスの流れ図である。ローカル・ストレージ106

【図1】

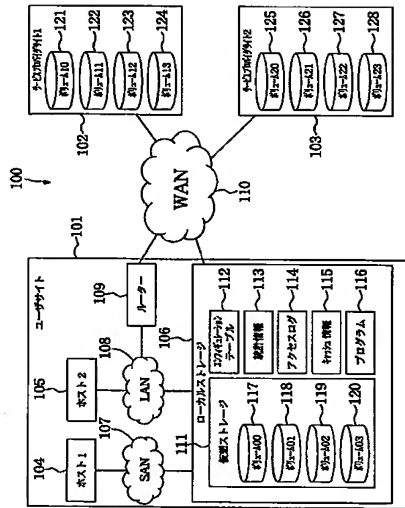
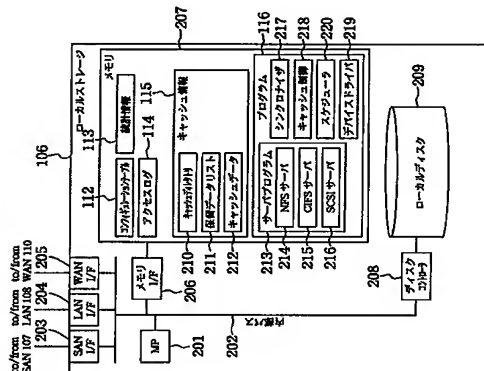


図 1

【図2】

図 2



【図6】

図 6

日付	時刻	コマンド	ファイルID	アドレス	サイズ
2000.0.1	00:00	取得済み	0	0x0000	0x0001
2000.0.1	00:00	書き込み	0	0x0000	0x0001
2000.0.1	00:01	取得済み	10	0x0100	0x0100
2000.0.1	00:01	書き込み	10	0x0100	0x0100
2000.0.1	00:02	取得済み	10	0x0200	0x0200
2000.0.1	00:02	書き込み	10	0x0200	0x0200
2000.0.1	00:02	取得済み	10	0x0300	0x0300
2000.0.1	00:02	書き込み	10	0x0300	0x0300

ータ・リスト211内のこの保留データの同期をとる必要があれば、シンクロナイザ217は保留データをサービス・プロバイダ・サイト(たとえば、サイト102)に送り、保留データとサービス・プロバイダ・サイトの同期をとる(ステップ904)。ローカル・ストレージ106内のキャッシュ制御218は、コンフィギュレーション・テーブル112(図4)内の「リモートID」及び「リモート・サイト」の値に基づいてサービス・プロバイダ・サイト(たとえば、サイト102)に格納すべきデータを認識する。

【0070】シンクロナイザ217は、保留データ・リスト211内の次のデータを選択する(ステップ905)。シンクロナイザ217は、保留データ・リスト211内で選択するデータがなくなるまでステップ902～905を繰り返す。保留データ・リスト211に選択するデータが上述のいずれかの方法を実行できるようにコンピュータで読み取り可能な媒体に格納できるプログラム又はコードを実装することも本発明の範囲である。

【0071】そこで、本発明について特定の実施形態を引用しながら説明してきたが、前記開示においては修正、さまざまな変更、及び置き換えを許容することが意図されており、規定されているとおり本発明の範囲を逸脱することなく他の特徴の対応する使用がなくても場合によっては本発明のいくつかの特徴を採用することは認められるであろう。

【図面の簡単な説明】

【図1】本発明の実施形態によるシステムのブロック図。

【図2】図1に示されているローカル・ストレージの実施形態の詳細を示すブロック図。

【図3】本発明の実施形態によるデータ・レイアウトの一例を示すブロック図。

【図4】本発明の実施形態によるコンフィギュレーション・テーブルの一例を示すブロック図。

【図5】本発明の実施形態による統計情報の一例のブロック図。

【図6】本発明の実施形態によるアクセス・ログの一例のブロック図。

【図7】本発明の実施形態による読み込みプロセスの流れ図。

【図8】本発明の実施形態による書き込みプロセスの流れ図。

【図9】本発明の実施形態による同期プロセスの流れ図。

【符号の説明】

101...ユーザ・サイト、102、103...サービス・プロバイダ・サイト、104、105...ホスト、106...ローカル・ストレージ、107...SAN、108...LAN、111...仮想ストレージ

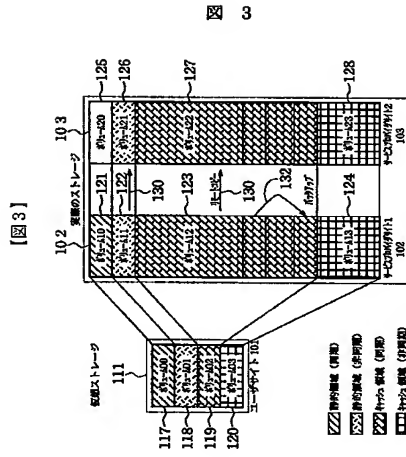
イブが同期領域の場合、ローカル・ストレージ106は即座にサービス・プロバイダ・サイト(たとえば、サイト102)とデータの同期をとる(書き込み) (ステップ808)、以下で説明するステップ809に進む。シンクロナイザ217(図2)はサービス・プロバイダ・サイトとの(ユーザ・サイト内の)データの同期処理を実行する。ローカル・ストレージ106内のキャッシュ制御218は、コンフィギュレーション・テーブル112を使用してサービス・プロバイダ・サイトに格納すべきデータを認識する。

【0065】ステップ806で、ボリューム・タイプが非同期領域の場合、ローカル・ストレージ106は保留データ・リスト211(図2)を更新する(ステップ807)。特に、サービス・プログラム(サブバ214、215、又は216)が保留データ・リストを更新する。【0066】続いてローカル・ストレージ106は、統計情報113(図5)を更新する(ステップ809)。特に、スケジューラ220は、統計情報113を更新する。統計情報113内のログ・ポイント(ログPtr)(図5)が設定されている場合、ローカル・ストレージ106はこの書き込みコマンドのログ・データをアクセス・ログ114に追加する。特に、スケジューラ220は書き込みコマンドのログ・データをアクセス・ログ114に追加する。これで書き込みプロセスは終了する。

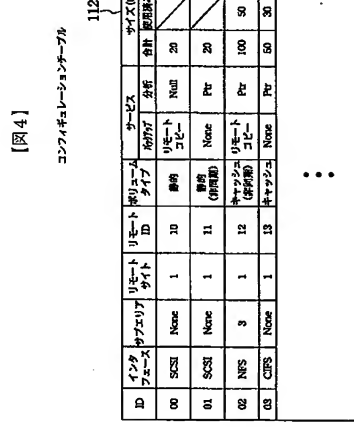
【0067】同期プロセス
非同期ボリューム書き込み(非同期領域へのデータ書き込み)の場合、ユーザ又はサービス・プロバイダによって定められたスケジューラに従ってサービス・プロバイダ・サイト(たとえば、サイト102)に送信される。このスケジューラは、たとえば、図4のコンフィギュレーション・テーブル112内の「ポリシー」エントリ内の値により定められる。スケジューラ220(図2)は、シンクロナイザ217(図2)を定期的に実行すること、サービス・プロバイダ・サイトとデータの同期をとることができ、この期間は、システム100設定に応じて、たとえば、約1.0ミリ秒又は10.0ミリ秒に設定できる。図9は、本発明の実施形態による同期プロセスの流れ図である。同期プロセスは、シンクロナイザ217(図2)によって実行できる。

【0068】シンクロナイザ217は、保留データ・リスト211(図2)内のヘッド・データを選択する(ステップ901)。最初に、シンクロナイザ217は、保留データ・リスト211にデータがなければ、シンクロナイザ217は同期プロセスを終了する。同期すべきデータが1つ又は複数(保留データ・リスト211内)にある場合、シンクロナイザ217は以下で説明するようにステップ902～905を実行する。

【0069】シンクロナイザ217は、保留データ・リスト211の情報を調べる(ステップ903)。保留データ・



☒ 3

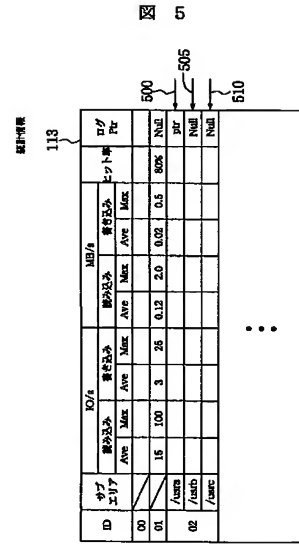


コンフィギュレーションチャート

ID	インデックス 番号	サブエリア 番号	リクエスト コード	リクエスト タイプ	リクエスト コメント	カーネル コメント	分析	合計 実行時間	サイズ (KB)	メモリー 使用量
00	SCSI	None	1	10	参照	リクエスト None	None	20		Null
01	SCSI	None	1	11	参照 (命令)	リクエスト None	Ptr	20		Null
02	NFS	3	1	12	リクエスト (参照)	リクエスト None	Ptr	100	50	置
03	CIFS	None	1	13	キャッシュ	None	Ptr	50	20	日

405

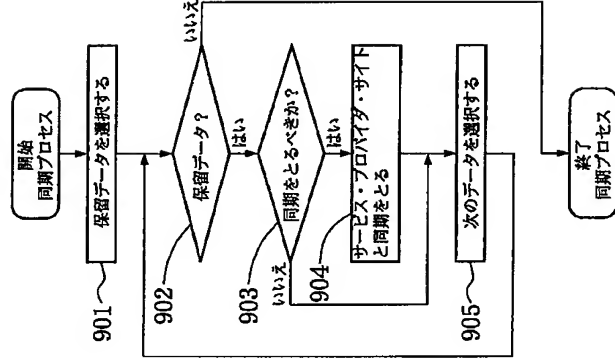
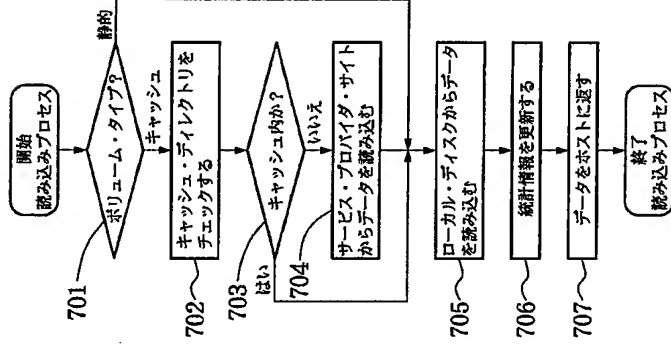
图 4



謝安

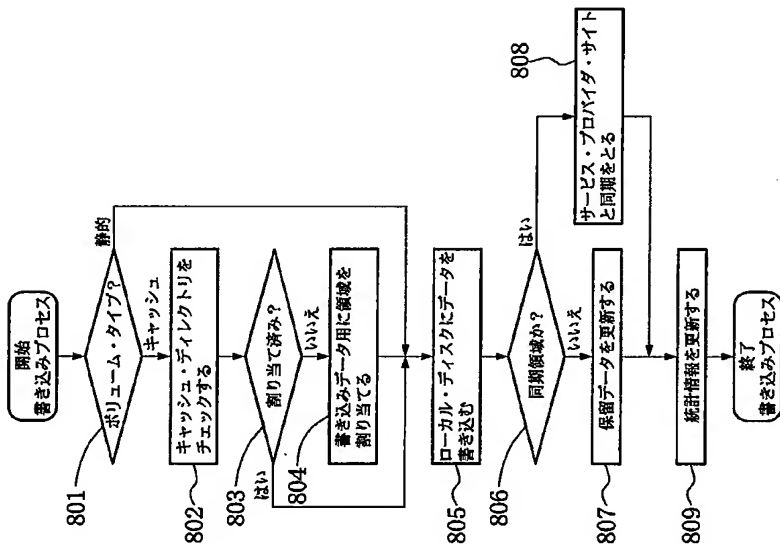
[illegible]

☒ 5



【図8】

8



フロントページの続き

Fターム(参考) 58065 CE21 EAS5 EK05
58082 FA11 FA12 G806 HA02

【外国語明細書】

1 Title of Invention

VIRTUAL VOLUME STORAGE

2 Claims

1. A system for providing a data storage service, the system comprising

: a service provider site configured to provide a data storage service

; and

a user site coupled by a wide area network (WAN) to the service prov

ider site, the user site comprising a local storage having a virtual sto

rage, the virtual storage having a synchronous volume and an asynchronou

s volume, the local storage configured to immediately transmit to the se

rvice provider site data that is written in the synchronous volume, to t

ransmit at a predetermined schedule to the service provider site data th

at is written in the asynchronous volume, and to read data from the serv

ice provider site if the data is not stored in the local storage.

2. The system of claim 1 wherein the local storage is configured to rec

ord an access activity to the local storage by the host.

3. The system of claim 1 wherein the local storage records an access ac

tivity to a volume in the virtual storage.

4. The system of claim 1 wherein the local storage records an access ac

tivity in a sub area of a volume in the virtual storage.

5. The system of claim 2 wherein the access activity is provided to the

service provider site for analysis.

6. A method of providing a data storage service, the method comprising:

providing a local storage having a virtual storage, the virtual stor

age comprising a synchronous volume and an asynchronous volume; and

selectively performing at least one of the following:

immediately transmitting to a service provider site data that is written in the synchronous volume of the virtual storage;

transmitting at a predetermined schedule to the service provider site data that is written in the asynchronous volume the virtual storage; and

reading data from the service provider site if the data is not stored in the local storage.

7. The method of claim 6, further comprising:
recording an access activity to the local storage.

8. The method of claim 7, further comprising:
providing a record of the access activity to the service provider site.

9. An electronically-readable medium storing a program for permitting a computer to perform a method of providing a data storage service, the method comprising:

selectively performing at least one of the following:

immediately transmitting to a service provider site data that is written in a synchronous volume of a virtual storage;

transmitting at a predetermined schedule to the service provider site data that is written in an asynchronous volume the virtual storage; and

reading data from the service provider site if the data is not stored in a local storage that includes the virtual storage.

10. A program code embedded on a carrier wave for causing a computer to perform a method of providing a data storage service, the method comprising:

selectively performing at least one of the following:

immediately transmitting to a service provider site data that is written in a synchronous volume of a virtual storage;

transmitting at a predetermined schedule to the service provider site data that is written in an asynchronous volume the virtual storage; and

reading data from the service provider site if the data is not stored in a local storage that includes the virtual storage.

11. An apparatus for processing data in a data storage system, the apparatus comprising:

a user site comprising a local storage having a virtual storage, the virtual storage including a synchronous area and an asynchronous area, the local storage configured to immediately transmit to a service provider site data that is written to the synchronous area, to transmit at a predetermined schedule to the service provider site data that is written to the asynchronous area, and to read data from the service provider site if the data is not stored in the local storage.

12. A method of processing data in a data storage system, the method comprising:

in response to a command signal to a local storage, selectively performing at least one of the following:

immediately transmitting to a service provider site data that is written to a synchronous area in a virtual storage;

transmitting at a predetermined schedule to the service provider site data that is written to an asynchronous area in the virtual storage;

reading data from the service provider site if the data is not stored in the local storage.

13. The method of reading data in a data storage system, the method comprising:

receiving a read command from a host to a local storage;

determining the volume type of the data that is requested by the host;

d command;
 if the volume type is static, then reading the data from a local disk in the local storage;
 if the volume type is cached, then checking if the data is stored in the local disk;
 if the data is stored in the local disk, then reading the data from the local disk;
 if the data is not stored in the local disk, then reading the data from a remote service provider site, storing the data in the local disk, and reading the data from the local disk;
 after reading the data from the local disk, updating statistical information relating to the read command; and
 returning the data to the host.

14. An apparatus for processing data in a data storage system, the apparatus comprising:
 a service provider site configured to immediately receive data that is written to a synchronous area in a virtual storage, to receive at a predetermined schedule data that is written to an asynchronous area in the virtual storage, and to transmit data to a local storage that includes the virtual storage if the data is not stored in the local storage.

15. A method of processing data in a data storage system, the method comprising:
 selectively performing one of the following:
 immediately receiving data that is written to a synchronous area in a virtual storage;
 receiving at a predetermined schedule data that is written to an asynchronous area in the virtual storage; and
 transmitting data to a local storage that includes the virtual storage if the data is not stored in the local storage.

16. A method of writing data in a data storage system, the method comprising:
 receiving a write command from a host to a local storage;
 determining the volume type of the data of the write command;
 if the volume type is static, then writing the data to a local disk in the local storage;
 if the volume type is cached, then checking if the data is stored in the local disk;
 if the data is stored in the local disk, then writing the data to the local disk from a cache;
 if the data is not stored in the local disk, then allocating a data area in the local disk and in the cache and then writing the data to the local disk from the cache;
 checking if the volume type of the data is synchronous;
 if the volume type is synchronous, then immediately synchronizing the data with a remote service provider site and then updating statistical information relating to the write command; and
 if the volume type is not synchronous, then synchronizing the data with a remote service provider site based on a predetermined schedule and then updating statistical information relating to the write command.

17. A system for providing a data storage service, the system comprising:
 means for selectively performing at least one of the following:
 immediately transmitting to a service provider site data that is written in a synchronous volume of a virtual storage;
 transmitting at a predetermined schedule to the service provider site data that is written in an asynchronous volume of the virtual storage;
 and
 reading data from the service provider site if the data is not s

tored in a local storage that includes the virtual storage.

3 Detailed Description of Invention

FIELD OF THE INVENTION

The present invention relates to data storage systems, and relates more particularly to a system and method for providing a virtual volume of data storage in a user site.

BACKGROUND OF THE INVENTION

Managing a large data storage system is very difficult. Typically, a data storage service provider provides a storage-related service such as providing storage volumes, data backup, and the like. From a user site, if a user connects to this storage via the storage service provider's directory, then the user must use a long distance connection to connect to this storage. This process causes more delays than if a user is connecting to a local storage.

The World Wide Web (WWW) effectively acts as a storage system in a wide area. The user may deploy a proxy server in the user site. As known to those skilled in the art, a proxy server is a server that sits between a client application (such as a web browser) and a remote server. The proxy server provides a cache of items that are available on the remote servers. The proxy server intercepts all requests that are made to the remote server so that the proxy server can determine if it can instead fulfill the request. If the proxy server is unable to fulfill the request, then the proxy server will forward the request to the remote server for processing. A proxy server can just only read cache data from the remote server or site, and it does not support a write procedure to the remote server or site. Additionally, the proxy server can not provide a service that depends on the data feature. As a result, the proxy server

er causes an ineffective usage of resources, and the use of a proxy server also causes more delays than the use of a local storage.

U.S. patent 5,155,935 discloses a multilevel, hierarchical, dynamically mapped data storage subsystem. This patent reference does not disclose storage systems between wide area networks (WANs) and ignores data features such as access interface type (block or file, and the like), purpose (user data or system file or backup file and the like). Thus, the subsystem disclosed in this reference is inefficient.

The Internet protocol small computer system interface (iSCSI) uses the Internet Protocol (IP) networking infrastructure to quickly transport large amounts of block storage (SCSI) data over existing local area and/or wide area networks. With the potential to support all major networking protocols, iSCSI (IP SAN) can unify network architecture across an entire enterprise, thereby reducing the overall network cost and complexity. To ensure reliability, iSCSI can use known network management tools and utilities that have been developed for IP networks. The iSCSI protocol is discussed, for example, at the website, <http://www.ece.cmu.edu/~ips/01ips/index.html> in the IP Storage section, <http://www.ece.cmu.edu/~ips/01ips/04ec.html>.

The working group of Internet Engineering Task Force (IETF) at <http://www.ietf.org> provides a network file system (NFS) version 3 (rfc1813 of IETF) and a common Internet File System (CIFS) protocol (<http://www.cifs.org>).

The NFS is an open operating system that allows all network users to access shared files that are stored in different types of computers. NFS provides access to shared files through an interface called Virtual File System (VFS) which runs on top of the Transmission Control Protocol/Internet Protocol (TCP/IP). With NFS, computers connected to a network can operate as clients while accessing remote files and as servers while

providing remote users access to local shared files.

The CIFS protocol defines a standard for remote file access using millions of computers at a time. With CIFS, users with different platforms and computers can share files without having to install new software. CIFS runs over TCP/IP, but uses the Server Message Block (SMB) protocol found in Microsoft Windows for file and printer access. Therefore, CIFS will allow all applications (including Web browsers) to open and share files across the Internet.

The websites <http://www.cit.com> and <http://www.ssa.com> describe wide area network (WAN) connections to a storage or storage area networks (SANs). The WAN connection may be an asynchronous transfer mode (ATM), a synchronous optical network (SONET), and the like.

The above references do not disclose methods for providing hierarchical management techniques between storage systems and host devices.

There is a need for a system and method that will overcome the above mentioned deficiencies of conventional methods and systems. There is also a need for a system and method that will permit an access service provider to have a reliable storage system and that will permit a user to quickly access the storage system. There is also a need for a system and method that will permit an access service provider to be able to tune a local storage system based upon user statistic data and user log data.

SUMMARY

The present invention may advantageously provide a reliable virtual local storage in a user site. The present invention may also advantageously provide a virtual local storage that permits faster access for a user. The present invention may also advantageously permit an access service provider to tune up the local storage by using user statistics data and user log data that are tracked by the local storage, and the service provider

may then charge the user for these tuning services.

In one embodiment, the present invention permits a service provider to provide a data storage service via a wide area network. A virtual volume storage is deployed at the user site. The virtual volume storage allows a host(s) at the user site to access the service provider storage located at the service provider site. The virtual volume storage advantageously permits the user to avoid having to access each time the wide area network coupled between the user site and the service provider site.

In one embodiment, the present invention broadly provides a system for providing a data storage service, comprising: a service provider site configured to provide a data storage service; and a user site coupled by a wide area network (WAN) to the service provider site, the user site comprising a local storage having a virtual storage, the virtual storage having a synchronous volume and an asynchronous volume, the local storage configured to immediately transmit to the service provider site data that is written in the synchronous volume, to transmit at a predetermined schedule to the service provider site data that is written in the asynchronous volume, and to read data from the service provider site if the data is not stored in the local storage.

In another embodiment, the present invention provides a method of managing data in a local storage based on the data feature. The virtual volume storage in the local storage determines if the data is static or cached, and synchronous or asynchronous. The virtual volume storage can also manage data not only by storage volume, but also by directory, file, cylinder, and/or block address.

In another embodiment, the present invention provides a method of tracking the access activities of a user in the user site. The user access activities are recorded to permit the service provider to analyze patterns in the user access activities. Based on this analysis, the service

provider can tune the virtual volume storage in the user site.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

The following description is provided to enable any person skilled in the art to make and use the present invention, and is provided in the context of a particular application and its requirements. Various modifications to the embodiments will be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other embodiments and applications without departing from the spirit and scope of the present invention. Thus, the present invention is not intended to be limited to the embodiments shown, but is to be accorded the widest scope consistent with the principles, features, and teachings disclosed herein.

In one embodiment in accordance with the present invention, a system 100 is configured by a user site 101, and by a first service provider site 102 and a second service provider site 103. The details of the user site 101 is described below. The number of service provider sites in the system 100 may vary. The first service provider site 102 includes storage volumes 121, 122, 123, and 124, while the second service provider site 103 includes storage volumes 125, 126, 127, and 128. The number of storage volumes in a service provider site may vary. The service provider sites 102 and 103 may be located in, for example, two different safe buildings. The user site 101, first service provider site 102, and second service provider site 103 are each connected to a Wide Area Network (WAN) 110. The service providers 102 and 103 provide data storage services via the WAN 110 to a user at the user site 101.

As known to those skilled in the art, a WAN is a computer network that typically spans a relatively large geographical area. A WAN may also

include local area networks (LANs). Computers connected to a WAN are often connected through public networks, such as the telephone system. They can also be connected through leased lines or satellites. The largest WAN in existence is the Internet.

The WAN 110 permits communication between each site (user site 101 and service provider sites 102 and 103). Each site 101, 102, and 103 may be separated from each other by long distances. The WAN 110 is typically configured by asynchronous transfer mode (ATM), synchronous optical network (SONET), Dense Wavelength Division Multiplexing (DWDM), or Internet Protocol (IP) network.

At the user site 101, a first host 104 is connected to a local storage 105 via a storage area network (SAN) 107, while a second host 106 is connected to the local storage 106 via a local area network (LAN) 108. The hosts 104 and 105 access the local storage 106. The hosts 104 and 105 are, for example, servers. The number of hosts in the user site 101 may vary. The LAN 108 is connected to the WAN 110 via, for example, a router 109. From the teachings of the present invention herein, it is understood by those skilled in the art that the user site 101 may be implemented to include only one of the SAN 107 or LAN 108, or both the SAN 107 and LAN 108.

As known to those skilled in the art, a SAN is a high-speed sub-network of shared storage devices wherein the SAN makes all storage devices available to all servers in a LAN or WAN. As more storage devices are added to a SAN, these added storage devices will also be accessible from any server in the larger network.

The SAN 107 may be configured with, for example, fibre channel or Small Computer System Interface (SCSI). As known to those skilled in the art, a fibre channel is a serial data transfer architecture developed by a consortium of computer and mass storage device manufacturers and now

being standardized by the American National Standards Institute (ANSI).

The most prominent fibre channel standard is the Fibre Channel Arbitrated Loop (FC-AL) which is designed for new mass storage devices and other peripheral devices that require a very high bandwidth. Using optical fibers to connect the devices, FC-AL supports full-duplex data transfer rates of approximately 100 megabytes per second (Mbps).

As also known to those skilled in the art, SCSI is a parallel interface standard used by Apple Macintosh computers, personal computers (PCs), and many UNIX systems for attaching peripheral devices to computers.

SCSI interfaces provide for faster data transmission rates (up to about 80 MBps) than standard serial and parallel ports.

As also known to those skilled in the art, a LAN is a computer network that typically spans a relatively small area. Most LANs are confined to a single building or group of buildings. Most LANs connect workstations and personal computers. Each node (individual computer) in a LAN has its own central processing unit (CPU) with which it executes programs, but it is also able to access data and devices anywhere on the LAN. Thus, many users can share expensive devices, such as laser printers, as well as data. Users can also use the LAN to communicate with each other, by, for example, sending e-mail or engaging in chat sessions. There are many different types of LANs, with Ethernets being the most common for PCs. LANs are capable of transmitting data at very fast rates, much faster than the data transmitted over a telephone line. However, the distances over LANs are limited, and there is also a limit on the number of computers that can be attached to a single LAN.

In Figure 1, the LAN 108 is configured as, for example, an Ethernet.

The first host 104 accesses the local storage 106 by use of, for example, a block (SCSI) interface (I/F). The second host 105 accesses the local storage 106 by, for example, a file (NFS or CIFS) I/F. The second

host 105 may also access the local storage 106 by, for example, a block (SCSI) I/F.

The local storage 106 and service provider sites 102 and 103 may use SCSI or fiber channel over ATM, fiber channel over SONET, or a unique transfer protocol.

The local storage 106 includes a virtual storage 111, configuration table 112 (see also Figure 4), statistical information 113 (see also Figure 5), access log 114 (see also Figure 6), cache information 115, and programs 116. The components 112, 113, 114, 115, and 116 permit the local storage 106 to act like a cache. The hosts 104 and 105 can access the virtual storage 111. The virtual storage 111 includes some volumes 117, 118, 119 and 120. The number of volumes in the virtual storage 111 may vary. These volumes 117-120 are managed by the local storage 106. The local storage 106 and service provider sites 102 and 103 collaborate in creation of the virtual volume storage 111.

The user can deploy the virtual volume storage 111 (in local storage 106) at the user site 101. The virtual volume storage 111 allows the users of host 104 and/or host 105 to access the huge service provider storage (volumes 121-124 and/or volumes 125-128) as the virtual volume storage 111 is deployed at the user site 101. The virtual volume storage 111 typically has a smaller size than the service provider storage volumes (volumes 121-124 and/or volumes 125-128). The virtual volume storage 111 has several interfaces such as, for example, a block I/F (SCSI), file I/F (NFS, CIFS) as discussed below. The virtual volume storage 111 advantageously permits the user to avoid having to access the WAN 110 every time. The virtual volume storage 111 also enables the service provider to provide a faster and more reliable storage system for the user.

Figure 2 is a block diagram showing the details of one embodiment of

the local storage 106. The local storage 106 includes a microprocessor 201 (which is available from, for example, Intel Corporation or Motorola Corporation), a local (internal) bus 202, a SAN interface (I/F) 203 coupled to the SAN 107 (Figure 1), a LAN I/F 204 coupled to the LAN 108 (Figure 1), a WAN I/F 205 coupled to the WAN 110 (Figure 1), a memory I/F 206, a memory 207, a disk controller 208, and a local disk 209. The memory 207 contains the configuration table 112, statistical information 113, access log 114, cache information 115, and programs 116. The microprocessor 201 controls all resources in the local storage 106 and executes all procedures in the local storage 106 by using the programs 116.

Figure 2 shows a snapshot of the local disk 209 during the running of procedures in the local storage 106. As known to those skilled in the art, a snapshot is a copy of a whole volume or a part of a volume of a real storage. These information and programs shown in Figure 2 are stored in the local disk 209. At the next sequence of local storage 106, these data and programs are moved from the local disk 209 to the memory 207.

The cache information 115 is configured with cache directory 210, pending data list 211, and cache data 212. The cache directory 210 is a directory (configuration information) of the cached data 212 in memory 207 and local disk 209. This cache data 212 may be managed by the least recently used (LRU) management. But some cases should be an exception of the LRU management, because there will be a possibility to access again in the near future. One such exception is sequential access where the storage address is accessed in a sequential manner.

The pending data list 211 is a list of pending data which is saved in the local storage 106. The pending data list 211 has information of each pending data such as a pointer to pending data, a pointer to a service provider site, and a synchronous period. A synchronous period is defined

by the data feature. If the data is not important, then this data may be stored in the local storage 106 for a predetermined period. This period may be, for example, one minute, one hour, one day, one week, or one month. If the data is important, then after the data is stored in the virtual storage 111, the data should be stored immediately (synchronously) in a service provider site (e.g., site 102) by using the configuration table 112 (Figure 4). For example, synchronous data is stored in Volume 00 117. In Figure 4, Volume 00 117 is identified with the identification number ID 00. Since the data stored in Volume 00 117 is synchronous data, the data will be immediately stored in the service provider site 102 (Remote Site 1) at Volume 10 121 (Remote ID 10).

This synchronous access feature provides a reliable data storage system. However, synchronous access requires much time because of the access time by the user site 101 to the service provider sites 102 and/or 103. The required access time depends on the particular operation being carried out on the system 100.

Data that are accessed by hosts 104 and/or 105 are stored in the memory 207. This data is called cache data 212. Cache data 212 is a temporary data that is frequently accessed by a user of host 104 and/or host 105.

In one embodiment, the programs 116 include server programs 213, a synchronizer 217, a cache control 218, a scheduler 220, and a device driver 219. The server programs 213 permit the local storage 106 and the host 104 (and/or host 105) to communicate with each other. The server programs 213 are configured with an NFS server 214, a CIFS server 215, and an SCSI server 216. The NFS server 214 acts as a conventional NFS server. The CIFS server 215 acts as a conventional CIFS server. The SCSI server 216 acts as a SCSI target device. The synchronizer 217 synchronizes the local storage 106 and the service provider sites 102 and/or 103 by

using the pending data list 211. The cache control 218 controls cache 212 in the memory 207 and local disk 209 by using the cache directory 210. The scheduler 220 schedules all procedures such as the server programs 213, synchronizer 217, cache control 218, and device driver 219. The device driver 219 controls all devices in local storage 106 such as the LAN I/F 203, the LAN I/F 204, the WAN I/F 205, the memory I/F 206, and the disk controller 208.

Figure 3 is a block diagram showing one example of a data layout in accordance with an embodiment of the present invention. The management of the local storage 106 is based upon the use of data feature, as described below in additional details. The virtual volume storage 111 deals with the data feature, depending on whether the data is static or cached, synchronous or asynchronous. As also described farther below, the virtual volume storage 111 can deal with data feature not only by storage volumes, but also by directory, file, cylinder, and/or block address. At user site 101, there are four (4) types of virtual volumes (Volume 00 117, Volume 01 118, Volume 02 119, and Volume 03 120) in the virtual storage 111. The features of these virtual volumes are described below: (1) the static synchronous volume, (2) the static asynchronous volume, (3) the cached synchronous volume, and (4) the cached asynchronous volume.

(1) The static synchronous volume (Volume 00 117):

Volume 00 117 is a static synchronous volume. Thus, Volume 00 117 occupies the same size as the real storage volume, Volume 10 121, in the first service provider site 102. The hosts 104 and/or 105 (at user site 101) read the data from Volume 00 117 and not from Volume 10 121. If the accessed data is in the cache data cache 212 (Figure 2), then the local storage 106 (Figure 2) does not access the local disk 209 (Figure 2)

and just accesses the memory 207. The hosts 104 and/or 105 write data to both Volume 00 117 and Volume 10 121. This static synchronous volume, Volume 00 117, provides a fast read access and reliable storage system.

(2) The static asynchronous volume (Volume 01 118):

Volume 01 118 is a static asynchronous volume. Thus, Volume 01 118 occupies the same size as the real storage volume, Volume 11 122, in the first service provider site 102. The hosts 104 and/or 105 read data from just only Volume 01 118 and not from Volume 11 122. If the accessed data is in the cache data 212, then local storage 106 does not access the local disk 209 and only accesses the memory 207. The hosts 104 and/or 105 write data to Volume 01 118 and register to the pending data list 211 (Figure 2). This pending data is synchronized with a predetermined schedule, as described below.

This pending data may be compressed in the background job in the local storage 106 (Figure 2) to reduce the WAN 110 traffic. This static asynchronous volume, Volume 01 118, provides a fast read and write volume, but may not provide the same reliability as a synchronous volume.

(3) The cached synchronous volume (Volume 02 119):

Volume 02 119 is a cached synchronous volume. Thus, Volume 02 119 occupies the same size as the real storage volume, Volume 12 123, in the first service provider site 102. Volume 02 119 acts as a cache. Thus, there is no data in Volume 02 119 that is accessed by the hosts 104 and/or 105. If the data is not in the local disk 209 (Figure 2), then the local storage 106 (Figure 2) reads data from the first service provider site 102 and writes the read data to the local disk 209. After the data is in the local storage 106, the local storage 106 sends the data from the local disk 209 via memory 207 to the hosts 104 and/or 105. If the accessed

d data is in the cache data 212, then the local storage 106 does not access local disk 209 and just accesses the memory 207. The hosts 104 and/or 105 write data to Volume 02 119 and Volume 12 123. This cached synchronous volume, Volume 02 119, advantageously permits the reduction of size of the local storage 106. This virtual volume, Volume 02 119, also provides a reliable storage system.

(4) The cached asynchronous volume (Volume 03 120):

Volume 03 120 is a cached asynchronous volume. Thus, Volume 03 120 occupies same size as the real storage volume, Volume 13 124, in the first service provider site 106. Volume 04 120 acts as a cache. Thus there is no data in Volume 03 120 that is accessed by the hosts 104 and/or 105. If the data is not in the local disk 209, the local storage 106 reads data from the first service provider site 102 and writes this read data into the local disk 209. After the data is already in the local storage 106, the local storage 106 sends the data from local disk 209 via memory 207 to the hosts 104 and/or 105. If the accessed data is in the data cache 207, then the local storage 106 does not access local disk 209 and just accesses the memory 207. The hosts 104 and/or 105 write the data to Volume 01 118 and register to pending data list 211 (Figure 2). This pending data is synchronized with a schedule. This cached asynchronous volume, Volume 03 120, advantageously permits the reduction of size of the local storage 106, but may not provide the same reliability as a synchronous volume.

Figure 3 also illustrates a service provided by the service provider site. Volume 11 122 in the first service provider site 102 is duplicated on volume 21 126 in the second service provider site 103. Volume 12 123 is duplicated on volume 22 127 in the second service provider site 103. Volume 13 124 is duplicated on volume 23 128 in the second service

provider site 103. This duplication (remote copying 130) of volume data provides a disaster recovery method. The volume data may be duplicated by use of, for example, conventional remote mirroring technology. Additionally, at the first service provider site 102, a backup of the data in Volume 12 123 may be made (see arrow 132) without requiring the function of the user site 101. This backup service permits the user to reduce the workload of data back up.

As now discussed below, the virtual volume storage 111 can make trace data based upon access activities from the hosts 104 and/or 105. The virtual volume storage 111 can provide to the service provider this trace data for purposes of analyzing the user access pattern. After analyzing the user access pattern, the service provider can advantageously tune the virtual volume storage 111.

Figure 4 is a block diagram of an example of a configuration table 112 in accordance with an embodiment of the present invention. The configuration table 112 is configured with the following parameters: "ID", "Interface", "Sub Area", "Remote Site", "Remote ID", "Volume Type", "Service" ("Backup", "Analyze"), "size" ("Total", "Used", "Free"), and "Policy". ID indicates volume identification of a virtual volume in the local storage 105. For example, Volume 00 117 (Figure 3) has an ID 00. Interface indicates the virtual volume interface type such as SCSI, NFS, and CIFS. Sub area indicates the number of sub areas in the virtual volume. A sub area may be a certain area of block address, cylinders, file, and/or directory. Volume type may be different between each sub area. In this case in the example of Figure 4, each sub area has the same volume type. For example, Volume 00 117 (Figure 3) (with ID 00) is a static synchronous volume type as mentioned previously. Remote Site indicates an identifier of the remote site (e.g., the identifier of service provider

site 102). This may be the IP address of an http address. Remote ID indicates volume ID in a service provider site. For example, Volume 1012 in service provider site 102 has a remote ID 10. Volume type indicates a type of volume such as static or cached, synchronous or asynchronous. Service indicates a service provided by the service provider. Such as a backup service or an analyze service. When a service provider provides an analyze service, the service provider sets a pointer to a statistical information 113 (Figure 2) and creates statistical information (access log) 114 to obtain a data feature. If an analyze service is provided, then "Ptr" will be indicated in the Analyze section in the Configuration Table 112 as shown in the example of Figure 4. For example, Volume 02 119 (ID 02) in Configuration Table 112 has a pointer Ptr in the Analyze section (see row 400). This pointer for ID 02 (of Volume 02 119) points to ID 02 in the Statistical Information 113 in Figure 5. ID 02 has a Sub area (directory) "/usr2" (see row 500) with a pointer ("ptr") in the Log Ptr section. This pointer (ptr) points to the access log 114 which is an access log created for Volume 02 119 in the example of Figure 6.

It is further noted that in the example of Figure 5, the ID 02 (of Volume 02 119) includes a directory "/usrb" (see row 505) with a "Null" value (no pointer) in the Log Ptr section. Additionally, ID 02 includes a directory "/usrc" (see row 510) with a Null value (no pointer) in the Log Ptr section. Thus, no associated access logs 114 have been created for the directories /usrb and /usrc since these directories do not have pointers that point to an access log.

In the Configuration Table 112, Size indicates total memory size (Total) in gigabytes, used size (Used), and free size (Free). If a volume is not in the file system in local storage 106, then only the total size needs to be known. The parameter policy indicates a schedule for synchronization. For example, in row 410, the Volume 00 117 (with ID 00) has

a "null" value set in the Policy parameter in Figure 4, and this null value indicates that a synchronization schedule has not been set for Volume 00 117. Volume 02 119 (with ID 02) has a "week" value set in the Policy parameter, and this value indicates that the data in Volume 02 119 is synchronized once per week with data in a service provider site (e.g., site 102). Volume 03 120 (with ID 03) has a "day" value set in the Policy parameter, and this value indicates that the data in Volume 03 120 is synchronized daily with data in a service provider site (e.g., site 102).

The parameters in the Configuration Table 112 are set by setting the configuration data in the Configuration Table 112. Each storage subsystem has an associated Configuration Table.

Figure 5 is a block diagram showing an example of statistical information 113 in accordance with an embodiment of the present invention. When a user or service provider sets the Service-Analyze parameter to "Ptr" in the Configuration Table 112 (Figure 4), then the following will be enabled. The local storage 106 collects the user statistical access information of local storage 106 in this table of Statistical Information 113. The Statistical Information is configured with "ID", "Sub Area", "IO/s", and "MB/s" (Read Ave, Max, Write Ave, Max), "Hit Ratio", and "Log Ptr". The parameters ID and Sub area (in Figure 5) have the same meanings as ID and Sub area in Configuration Table 112 (Figure 4). IO/s indicates the number of host/user input or output activities per second such as read and write commands per second. A read or write access in a volume or in a volume directory may be recorded by use of any suitable known methods.

MB/s indicates number of bytes per second. The local storage 106 collects each read and write occurrence in a volume (or in a sub area in a

volume), and averages (Ave) this occurrences. The local storage 106 also tracks the maximum occurrence (Max) of read and write occurrence in a volume or in a sub area in a volume.

Hit Ratio indicates a hit ratio of local storage. This hit ratio is, for example, only just for the local disk 209 (Figure 2). A hit ratio is defined in equation (1):

$$(1) \text{ Hit ratio} = 100\%(\text{\#LOCAL DISK READ}/\text{\#ALL READS})$$

where #LOCAL DISK READ is the number of reads in the local disk 209 by a host, and #ALL READS is the number of total reads by the host.

Statistical information 113 is sent to the service provider sites 102 and/or 103 periodically. The service provider analyzes the feature of data, and from this analysis the service provider can propose better solutions to the user of a host at the user site 101.

As stated above, Log Ptr indicates a pointer to the access log 114.

If the service provider wants to know more detail of user access pattern as, then the service provider can set this pointer to a created accesses log 114.

Figure 6 is a block diagram showing an example of an access log 114 in accordance with an embodiment of the present invention. When the service provider wants to analyze the details of a user's access pattern, then the service provider creates an access log 114 and sets a pointer at Log Ptr (in Statistical Information 113 in Figure 5) to this access log 114. The local storage 106 collects an access log 114 of each access by a user. This access log 114 is configured with the parameters: "Date", "Time", "Command", "File ID", "Address", and "Size". Date indicates the date of an access by a user. Time indicates the time of an access. Command indicates the command type (e.g., read or write) of an access. File ID indicates the identification of a file that was accessed by this

command. If access log is for SCSI, the File ID may be null. Address and size indicates the access address and size.

Read Process

In the read procedure, the cached device (local storage 106) acts as cache. The hosts 104 and/or 105 access the local storage 106, since the local storage 106 has a large space. At first, a host (e.g., host 104 or host 105) issues a read command of NFS, CIFS, or iSCSI protocol via LAN 108 (Figure 1), or a read command of SCSI protocol via SAN 107. The local storage 106 receives the read command via LAN 1/F 204 or SAN 1/F 203 (Figure 2). The device driver 219 (Figure 2) handles this read command from a host and places this read command into the scheduler 220. The scheduler 220 then analyzes the read command and places the read command into a proper server (NFS server 214, CIFS server 215, or SCSI server 216) within the server programs 213 for purposes of processing the read command. Each server program 213 checks whether the data requested by the read command is in or not in the cache data 212 (Figure 2). If all data (which is requested by the read command) is in the cache data 212, then the local storage 106 returns data to the requesting host (host 104 or 105) via internal bus 202 and a network interface (SAN 1/F 203 or LAN 1/F 204). On the other hand, if all or part of the data (requested by the read command) is not in cache data 212, then the requested data should be moved to cache data 212 from the local disk 209 or from the service provider sites 102 or 103.

Figure 7 is a flowchart diagram of this read process in accordance with an embodiment of the present invention. After the local storage 106 receives a read command from a host (e.g., host 104 or 105), the local storage 106 checks 701 for the volume type of the data. The scheduler 220 checks configuration table 112 for the volume type (e.g., static type

) by check a volume ID. In particular, the scheduler 220 (Figure 2) analyzes the read command to determine the volume type of the data and to determine which server program (server 214, 215, or 216) should process (handle) the read command. The scheduler 220 looks at configuration table 112 (Figure 4) for the volume type and the appropriate server program to handle the read command. If volume type is a static volume, then the local storage 106 skips the steps 702, 703, and 704 and proceeds to step 705 which is discussed below. If the volume type is a cached volume, then the local storage 106 checks 702 if the data (requested by the read command) is stored in or not stored in the cache data 212.

The local storage 106 checks 703 the cache directory 210. In particular, the appropriate server (214, 215, or 216) in the server program 213 checks whether the data requested by the read command is in or not in the local disk 209. If all data (requested by the read command) is stored in the local disk 209, then the local storage 106 skips step 704 and proceeds to step 705 which is described below. If data (request by the read command) is not in the local disk 209, then local storage 106 executes step 704.

In step 704, the local storage 106 allocates data area in the local disk 209 and reads data (requested by the read command) from a service provider site (e.g., service provider site 102). In particular, the cache control 218 uses the configuration table 112 (Figure 4) to determine the service provider site from where data should be obtained. The cache control 218 reads the data from the service provider site and stores the data in the local disk 209.

After data is stored in local disk 209, the local storage 106 will move (read) 705 the data from the local disk 209 to the cache data 212.

In particular, the cache control 218 moves the data from the local disk 209 to the cache data 212. The local storage 106 then updates 706 the s

statistical information 113 (Figure 5). In particular, the scheduler 220 in the local storage 106 updates the statistical information 113. If the log pointer (Log Ptr) (Figure 5) in the statistical information 113 is set, then the local storage 106 adds the log data of this read command to the access log 114 (Figure 6). The scheduler 220 or the device driver 219 creates the statistical information 113 and access log 114. Typically, the scheduler 220 is preferred in creating the statistical information 113 and access log 114.

The data (requested by the read command) is then moved 707 from the cache data 212 to the host (e.g., host 104 or 105) that sent the read command. In particular, the cache control 218 moves the data from the cache data 212 to the appropriate server program (214, 215, or 216), and the appropriate server program (214, 215, or 216) returns the data to the host that sent the read command.

Write process

In the write procedure, the cached device (local storage 106) acts as a cache. The host 104 and/or host 105 access the local storage 106, since the local storage 106 has a large space. At first, a host (e.g., host 104 or host 105) issues a write command of NFS, CIFS, or iSCSI protocol via LAN 108, or a write command of SCSI protocol via SAN 107. The local storage 106 receives the write command via LAN I/F 204 or SAN I/F 203. The device driver 219 (Figure 2) handles this read command from a host and places this read command into the scheduler 220. The scheduler 220 then analyzes the write command and places the write command into a proper server (NFS server 214, CIFS 215, or SCSI server 216) within the server programs 213. Each server program 213 checks whether the data of the write command is in or not in the cache data 212 (Figure 2). If all data (of the write command) is in the cache data 212, then the local s

storage 106 receives the data from the host via internal bus 202 and a network interface (SDN I/F 203 or LAN I/F 204). The local storage 106 will order write the data on an allocated space in the local disk 209 and cache 212. On the other hand, if all or part of the data (of the write command) is not in the cache data 212, then the local storage 106 will allocate an area (in local disk 209 and cache data 212) for the rest of data not in the cache data 212. After all of the data area is allocated, then the local storage 106 stores data in that allocated area. When all data is stored in the cache data 212, then the local storage 106 stores data in local disk 209 and sends this data to a provider site (e.g., service provider site 102).

Figure 8 is a flowchart diagram of a write process in accordance with an embodiment of the present invention. After the local storage 106 receives a write command from a host, the local storage 106 checks 801 the volume type of the data of the write command. In particular, the scheduler 220 (Figure 2) analyzes the write command to determine the volume type of the data and to determine which server program (server 214, 215, or 216) should process (handle) the read command. The scheduler 220 looks at configuration table 112 (Figure 4) for the volume type and the appropriate server program to handle the write command. If the volume type is a static volume, then the local storage 106 skips steps 802, 803, and 804, and proceeds to step 805 which is described below. If the volume type is a cached volume, then the local storage 106 checks if the data (of the write command) is stored or not stored in cache data 212 (Figure 2).

The local storage 106 checks 802 the cache directory 210 (Figure 2). In particular, the appropriate server (214, 215, or 216) in the server program 213 checks whether the data of the write command is allocated in the local disk 209. In step 803, if all the data (requested by the wr

ite command) is allocated in the local disk 209, then the local storage 106 skips step 804 and proceeds to step 805. In step 805, if the data is not in the local disk 209, then local storage 106 executes step 804.

If the data (of the write command) is not allocated in the local disk 209, then the local storage 106 allocates 804 data area in both the local disk 209 and cache data 212 for the write data. In particular, the cache control 218 allocates the data area on both the local disk 209 and cache data 212.

The local storage 106 then writes 805 the write data to the local disk 209 via the cache data 212. In particular, the appropriate server program (NFS server 214, CIFS server 215, or SCSI server 216) writes the data to the local disk 209.

The local storage 106 checks 806 the volume type in which the data is written by use of the configuration table 112 (Figure 4). In particular, the server program (NFS server 214, CIFS server 215, or SCSI server 216) uses the configuration table 112 to check for the volume type. If the volume type is a synchronous area, then the local storage 106 immediately synchronizes (writes) 808 the data to a service provider site (e.g., site 102) and proceeds to step 809 which is discussed below. The synchronizer 217 (Figure 2) performs the synchronizing of the data (in the user site) to the service provider site. The cache control 218 in the local storage 106 knows where the data should be stored in the service provider site by use of the configuration table 112.

In step 806, if the volume type is an asynchronous area, then the local storage 106 updates 807 the pending data list 211 (Figure 2). In particular, the server program (server 214, 215, or 216) updates the pending data list.

The local storage 106 then updates 809 the statistical information 113 (Figure 5). In particular, the scheduler 220 updates the statistical

information 113. If the log pointer (Log Ptr) (Figure 5) in statistic information 113 is set, then the local storage 106 adds the log data of this write command to the access log 114. In particular, the scheduler 220 adds the log data of the write command to the access log 114. The write process then ends.

Synchronization Process

In the case of an asynchronous volume write (a data write to an asynchronous area), the data should be sent to a service provider site (e.g., site 102) by a schedule that is defined by the user or service provider. This schedule is, for example, defined by the value in the "Policy" entry in the configuration table 112 in Figure 4. The scheduler 220 (Figure 2) executes the synchronizer 217 (Figure 2) periodically to enable data synchronization with a service provider site. This period may be set to, for example, approximately 1.0 milli-second or 10.0 milli-seconds, depending on the system 100 setting.

Figure 9 is a flowchart diagram of synchronization process in accordance with an embodiment of the present invention. The synchronization process may be performed by the synchronizer 217 (Figure 2).

The synchronizer 217 selects 901 the head data in the pending data list 211 (Figure 2). At the first, the synchronizer 217 checks 902 the pending data list 211. If there is no data in the pending data list 211, then the synchronizer 217 ends the synchronization process. If there is one or more data (in the pending data list 211) which should be synchronized, then the synchronizer 217 executes steps 902-905 as described below.

The synchronizer 217 checks 903 the information of pending data list 211. If this pending data in the pending data list 211 should be synchronized, then the synchronizer 217 sends the pending data to a service p

rovider site (e.g., site 102), so that the pending data is synchronized 904 with the service provider site. The cache control 218 in the local storage 106 knows where data should be stored in the service provider site (e.g., site 102) based upon the values in the Remote ID and Remote Site in the configuration table 112 (Figure 4).

The synchronizer 217 then selects 905 the next data in the pending data list 211. The synchronizer 217 repeats steps 902 through 905 until there is no more data to select in the pending data list 211. If there is no more data to select in the pending data list 211, then the method of Figure 9 ends.

It is also within the scope of the present invention to implement a program or code that can be stored in an electronically-readable medium to permit a computer to perform any of the methods described above.

Thus, while the present invention has been described herein with reference to particular embodiments thereof, a latitude of modification, various changes and substitutions are intended in the foregoing disclosure, and it will be appreciated that in some instances some features of the invention will be employed without a corresponding use of other features without departing from the scope of the invention as set forth.

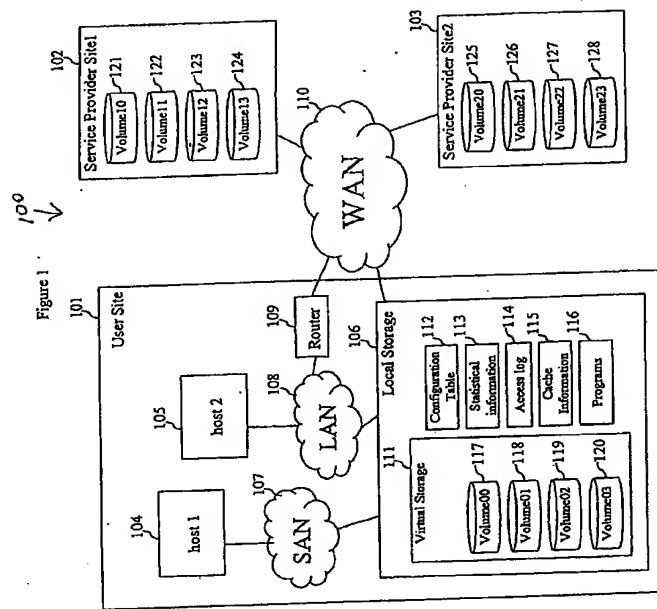
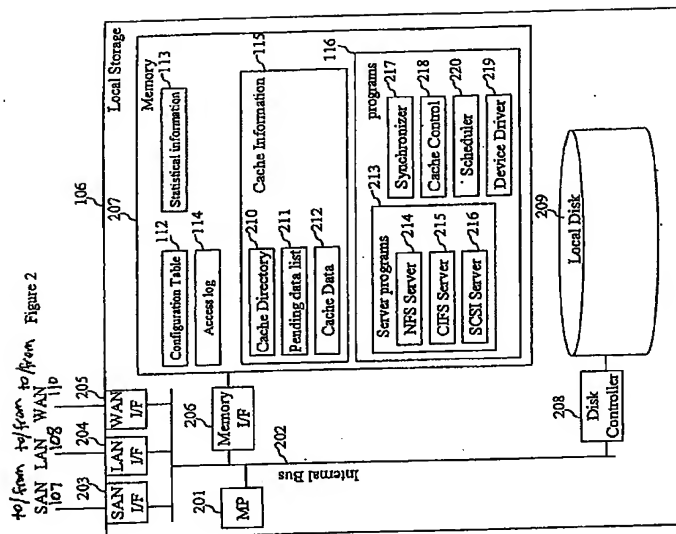
4. Brief Description of Drawings

Figure 1 is block diagram of a system in accordance with an embodiment of the present invention;

Figure 2 is a block diagram showing additional details of one embodiment of the local storage in Figure 1;

Figure 3 is a block diagram showing one example of a data layout in accordance with an embodiment of the present invention;

Figure 4 is a block diagram of an example of a configuration table i
n accordance with an embodiment of the present invention;
Figure 5 is a block diagram of an example of statistical information
in accordance with an embodiment of the present invention;
Figure 6 is a block diagram of an example of an access log in accord
ance with an embodiment of the present invention;
Figure 7 is a flowchart diagram of a read process in accordance with
an embodiment of the present invention;
Figure 8 is a flowchart diagram of a write process in accordance with
an embodiment of the present invention; and
Figure 9 is a flowchart diagram of synchronization process in accord
ance with an embodiment of the present invention.



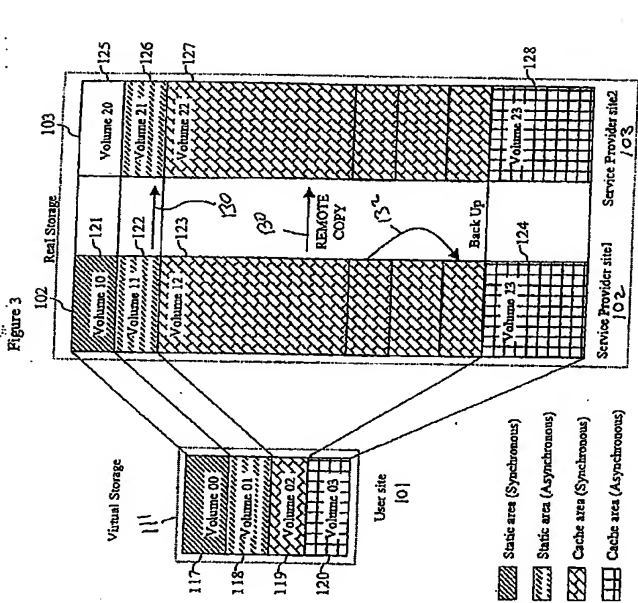


Figure 5

Statistical Information

ID	Sub Area	IOs				MB/s				Hit Ratio	Log Pr
		Ave	Max	Ave	Max	Ave	Max	Ave	Max		
00		15	100	3	25	0.12	2.0	0.02	0.5	80%	Null
01	Area										Pr
02	Area										Null

Figure 6

Access log 114

Date	Time	Command	File ID	Address	Size
2000.8.1	0:0:0	Read	0	0x0000	0x0001
2000.8.1	0:0:0	Write	0	0x0000	0x0001
2000.8.1	0:0:1	Read	10	0x0100	0x0100
2000.8.1	0:0:1	Read	10	0x0200	0x0100
2000.8.1	0:0:2	Read	10	0x0200	0x0100
2000.8.1	0:0:2	Read	10	0x0300	0x0100
2000.8.1	0:0:2	Read	10	0x0400	0x0100

Figure 7

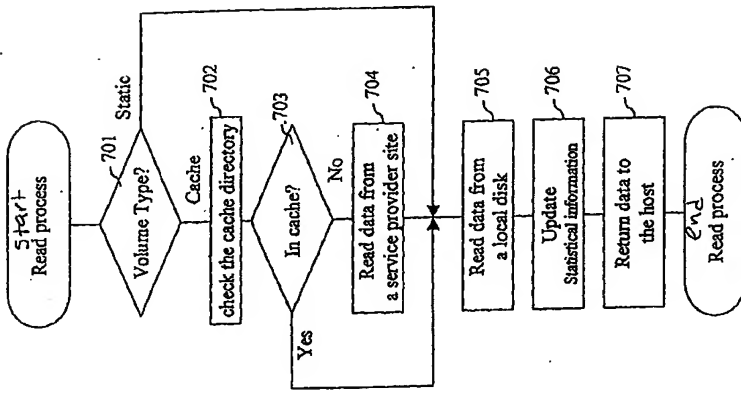


Figure 8

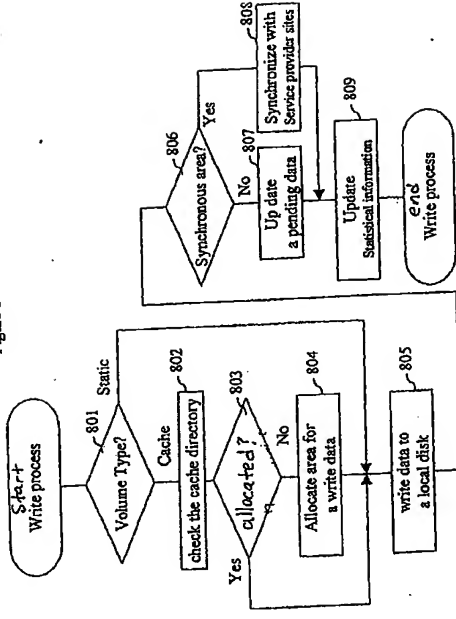
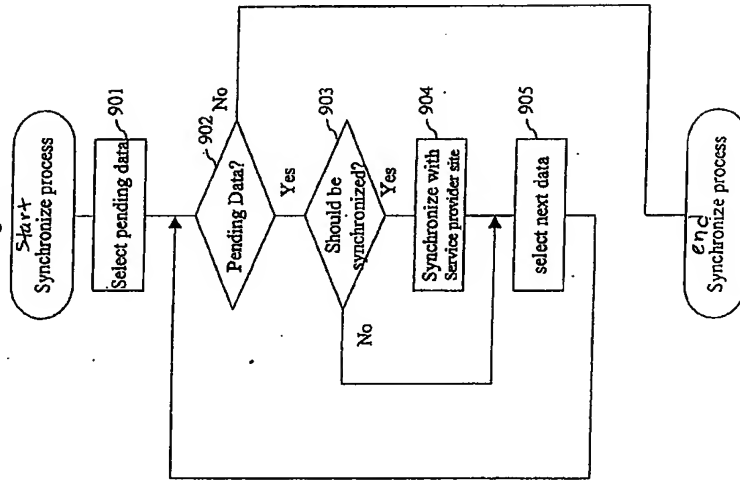


Figure 9



1. Abstract

A system for providing a data storage service, comprises: a service provider site configured to provide a data storage service; and a user site coupled by a wide area network (WAN) to the service provider site, the user site comprising a local storage having a virtual storage. the virtual storage having a synchronous volume and an asynchronous volume, the local storage configured to immediately transmit to the service provider site data that is written in the synchronous volume, to transmit at a predetermined schedule to the service provider site data that is written in the asynchronous volume, and to read data from the service provider site if the data is not stored in the local storage.

2. Representative Drawing

Fig. 1